Information Dynamics and Emergent Behavior of Heterogeneous-Agent Systems

GRANT NO.:	**F49620-99-1-0014**
PI:	**Nong Ye**
SPONSOR:	**AFOSR/DARPA**
REPORTING PERIOD:	**January 1, 2001 – December 31, 2001**

Table of Contents

List of Figure Captions

List of Table Captions

ABSTRACT

This project presents an effort to establish theories and techniques of modeling, analyzing and controlling information dynamics and emergent behavior of heterogeneous-agent systems and demonstrate the application of these theories and techniques to heterogeneous-agent systems making up information supply networks. A heterogeneous-agent system is a Complex Adaptive System (CAS) involving a large collection of semi-autonomous agents. According to Complexity Theory for CAS, the aggregate system behavior of heterogeneous agents and their inter-connections emerges from the evolving local interactions of agents in a dynamically changing environment. A CAS best organizes from the bottom up through self-organization. We base the modeling of heterogeneous-agent systems on the innovative integration of three modeling paradigms: agent-based modeling, control-theoretic modeling, and stochastic discrete-event modeling. The model represents the physical, information and knowledge elements of each agent, which includes: fitness function, tagging, building blocks, internal model, dynamic resources and processes under the control of the agent, as well as the interaction of the agent with other agents and the environment. The analysis of heterogeneous-agent systems and their emergent behavior is based on the innovative application of non-linear time-series analysis techniques from chaos theory and nonlinear dynamical systems theory as well as multivariate statistical techniques to detect emergent states and temporal patterns. We base the control of heterogeneous-agent systems on the innovative bottom-up self-organization approach to coping with desirable and undesirable emergent states of heterogeneous-agent systems for system stability and robustness. This project investigates the emergent behavior of information supply network systems (e.g., military information supply networks for information distribution and fusion in command and control – C2, and commercial information supply networks for e-commerce). In this project, we build and use the simulation model of an information supply network system to collect behavioral data of the system under experimental conditions involving various inter-agent control methods and different network environments. This project also develops and applies analytical techniques to the simulation data for understanding

emergent system behavior under various conditions and gaining insights into system design and control. The simulation experiments look into the following factors:

1) Inter-agent control methods for different degrees of agent autonomy and agent coordination, such as sharing of state information among agents, conformance to a common fitness function by agents, enforcement of the same internal model among agents and so forth, and

2) System structures determined by types and density of inter-connections among agents through material, information and knowledge flows.

INTRODUCTION

A supply network enterprise is a network of organizations coupled for the purpose of providing value to customers. A supply network enterprise consists of a focal organization and the network of firms that transact with it in the form of physical goods and services as well as information. These connections are then extended iteratively (i.e., connecting the supplier's suppliers and customers and the customer's customers and other suppliers). An enterprise potentially transcends multiple industries and markets. Each organization is an agent acting on its own self-interest, but subject to constraints from its contracting relationships with other agents and from the environment. Agents are heterogeneous in their role and other features in a supply network enterprise. A complex network of suppliers, manufacturers, distributors, transporters, retailers, and customers – a supply network enterprise – is common. The complexity of interrelationships arises from not only material flows but also information flows throughout the supply network enterprise. Little exists now in understanding the emergent behavior of supply network enterprises, designing their structures, and optimizing their operation.

A supply network enterprise is a heterogeneous-agent system – a Complex Adaptive System (CAS) – involving a large collection of semi-autonomous agents that may play different roles in the system. According to Complexity Theory for CAS, the aggregate system behavior of heterogeneous agents and their inter-connections emerges form the evolving, local interactions of agents in a dynamically changing environment.

Advances in information technology have led us to a new era of business organization and market structure. Information technology gives a focal organization in a supply network enterprise an unprecedented capability to communicate, coordinate, and even control its suppliers, distributors, transporters, retailers, customers and itself. For example, the focal organization may attempt to improve the delivery time, product quality and cost by obtaining the information of inventory and process quality from suppliers, the information of product sales from retailers, and even coordinating and controlling the business process throughout the supply network via Enterprise Resource Planning (ERP) software. The focal organization can even interact with customers directly to provide products and services and in turn, consumers can benefit by obtaining objective comparative information about their desired purchase.

Despite potential changes that can be introduced by information technology, it is still not clear what kinds of changes will produce the desirable emergent behavior of a supply network enterprise, and thus should be adopted. For example, the following questions remain to be answered:

- Should a focal organization exert tight control over its suppliers by monitoring and controlling the product development and production process of its suppliers and the inventory management process of its distributors, which is made possible by information technology, or should the focal organization leave a high degree of autonomy to its suppliers and distributors for innovation and flexibility?
- What kind of suppliers should a focal organization choose to work with, a few suppliers or a large number of suppliers, suppliers which commit all of their resources to support the focal organization for a long period of time, or suppliers which have more freedom to remain or leave the supply network enterprise dynamically?
- How should a focal organization deal with other agents in the supply network enterprise when facing an unstable market with many changes, versus a stable market?
- Which inter-agent control methods and which supply network structures, under which nature of internal dynamics and eternal changes, leads to the desirable emergent behavior of a supply network enterprise?

The long-term goal of this research work is to establish theories and techniques of modeling, analyzing and controlling heterogeneous-agent systems and their emergent behavior, and to demonstrate the application of these theories and techniques to supply network enterprises in e-business. The work on the simulation modeling of heterogeneous-agent systems will focus on a coherent integration of three modeling paradigms: agent-based modeling, control-theoretic modeling and stochastic process modeling, to represent essential features of agents such as the role, internal model, building blocks, fitness function, physical and information flows, aggregation, diversity and non-linearity. The work on the analysis of heterogeneous-agent systems and their emergent behavior will focus on non-linear time-series analysis techniques from chaos and dynamical systems theories and multivariate statistical techniques to detect emergent states and temporal patterns. The work on the control of heterogeneous-agent systems will focus on the bottom-up self-synchronization approach to handling desirable and undesirable emergent states of heterogeneous-agent systems for system stability and robustness according to Complexity Theory for CAS.

This project report represents the objectives of the initial one-year effort:

1) Build and investigate a simulation model of a supply network enterprise – a heterogeneous-agent system – under experimental conditions involving various inter-agent control methods, internal dynamics, and external changes, and

2) Develop and apply analytical techniques to the simulation data for understanding the emergent system behavior under various conditions and gaining insights into system design and control.

Findings from this effort will provide insights into supply network management (e.g., selection and management of suppliers) as well as insights into the emergent behavior of heterogeneous-agent systems in general. This effort will also demonstrate the feasibility and validity of our simulation modeling approach and analytical methodology for the understanding of heterogeneous-agent systems.

After this one-year effort, we plan to investigate control strategies handling desirable and undesirable emergent states of CAS for the performance, stability, and robustness of CAS.

Chapter 1: Complex Adaptive Systems

Recent efforts have shown that due to the complex, dynamic nature of a supply network, it is not enough to model the network as a mere system. More appropriately, a supply network should be modeled as a Complex Adaptive System (CAS) [1]. Therefore, in our experiments, we model the supply chain enterprise as a heterogeneous-agent CAS.

1-1 Complex Adaptive System Theory

A heterogeneous-agent system is a CAS involving a large collection of semi-autonomous agents that may play different roles in the system. Within each agent is a set of states and behaviors that can be used to describe the agent. Each agent works independently to increase the level of fitness achieved by the agent as well as the surrounding regional or global network of which it is a member. Agents within the CAS are allowed measurable degrees of freedom to behave in a semi-autonomous fashion. The degree of freedom that is afforded an agent is determined by the dimensionality of the CAS. [1] By reducing the dimensionality (through control techniques) or increasing the dimensionality by simply allowing the agents a higher degree of autonomy, we can decrease or increase the level of stochastic behavior present in the CAS respectively. This technique allows us to observe forced behavior under tight control, versus the emergent behavior of a less restricted network.

A key concept in CAS Theory is the connectivity of the agents within the CAS. The number of connections within the system, as well as the characteristics (including level of communication) of each connection dictates the dynamics of the system's communication capabilities. These connections between agents form multiple relationships of varying degrees. The interrelations formed by connectivity are indicative of the network's potential to "engage in global communication from within" [1].

An interactive relationship is present between a CAS and the external environment in which it exists [1]. According to Complexity Theory for CAS, the aggregate system behavior of heterogeneous agents and their inter-connections emerges from the evolving, local interactions of agents in a dynamically changing environment. Therefore, the observed behavior in a CAS is not governed by a single entity, but by the simultaneous actions of the agents within the system, as well as the co-evolution of both the system itself and its environment [1]. A CAS is thus considered a self-organizing entity with observable emergent behavior.

1-2 Supply Network Enterprise

A supply network enterprise is a network of organizations coupled for the purpose of providing value to a customer. A supply network enterprise consists of a focal organization and the network of firms that transact with it in the form of physical goods and services as well as information. These connections are then extended iteratively (i.e., connecting the supplier's suppliers and customers and the customer's customers and other suppliers). An enterprise potentially transcends multiple industries and markets. Agents in a supply network enterprise are subject to varying degrees of connectivity with other agents [1]. This structure determines the availability of information flow through the network.

Each organization is an agent acting on its own self-interest, but subject to constraints from its contracting relationships with other agents and from the environments. Agents are heterogeneous in their role and other features in a supply network enterprise. Complex networks of suppliers, manufacturers, distributors, retailers, and customers – an enterprise – are commonplace in many industries, including E-Commerce. The complexity of interrelationships arises from not only material flows, but also information flows throughout the enterprise. Little exists now in understanding the emergent behavior of supply network enterprises, designing their structures, and optimizing their operation.

The theory of Complex Adaptive Systems seems to fit naturally into the description of a supply network enterprise. Due to this observation, we model our supply network enterprise as a CAS.

Chapter 2: Modeling a Supply Network Enterprise

The system modeling will be based on a coherent integration of three modeling paradigms: agent-based modeling, control-theoretic modeling, and stochastic process modeling -as shown in the following figure. This structure applies the comprehensive cohesion of numerous current techniques used in modeling supply network enterprises. [2-6]

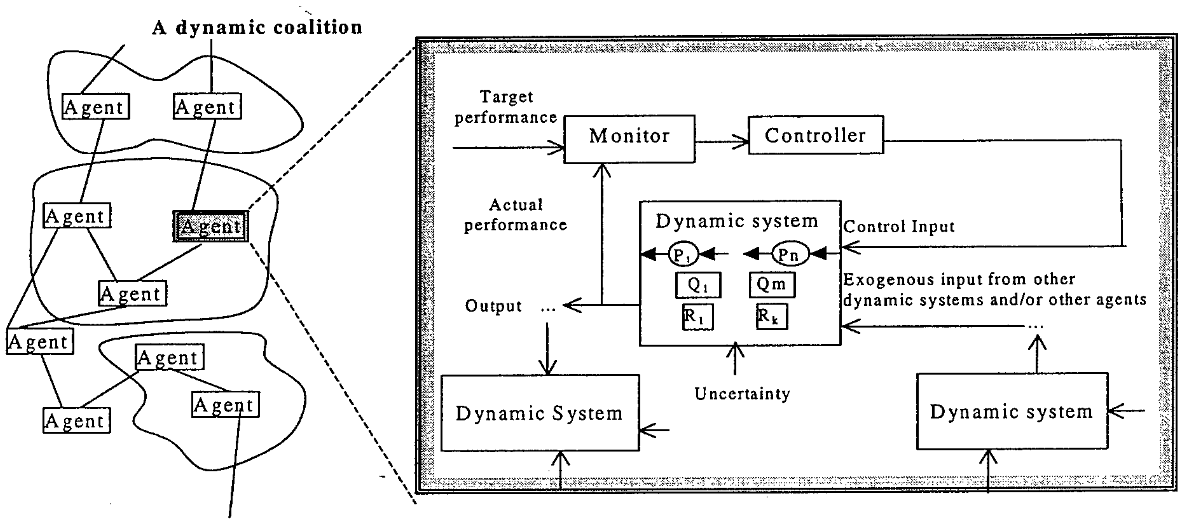

Figure 2-1: Complex adaptive system model of a supply network enterprise.

Control-theoretic modeling based on control theory will be used to specify the proactive, negative-feedback logic of an agent in the controller of the agent for the adaptive, unexpected behavior. Stochastic process modeling based on queuing theory and stochastic process theory will be used to specify dynamic systems and uncertainties in the environment from which agents gain information and on which agents produce impact.

We use these three functions, which are described, in section 2-2.2 in the following sections to describe agent fitness and production calculations:

gap(i) =	*production(i) - demand(i)*
customerGap(i) =	*production(i) - demand(customer)*
endGap(i) =	*production(i) - demand(endCustomer)*

2-1 Agent Design

Each agent will contain the following elements: role, fitness function, internal model, building blocks, dynamic system controlled by the agent, and connectivity with other agents and the environment. At any time, the agent's state can be defined by the state of each of its elements.

2-1.1 Role of Agent

The role of an agent indicates the type of the agent. An agent's type can be a focal organization, supplier, or customer. The agent's role in the system depends on its type. Suppliers are measured by their distance from the focal organization, for example, suppliers that supply directly to the focal organization are 1-tier suppliers, their suppliers are 2-tier suppliers, and so on. For every supplier, the supplier's customer exists at a tier one level above (numerically lower than) the supplier. This structure enables us to observe sub-structures within the network. For example, a supplier, along with its immediate suppliers and customer, forms a regional network that exists within the global supply network enterprise where the supplier acts as the focal organization for the smaller sub-network.

2-1.2 Fitness Function

The fitness function describes the objective of the agent, and the relationship of the objective measure (e.g., profit), with various factors. These factors include: the state of the dynamic system (e.g., the inventory level), the output performance (e.g., number of finished orders, delivery time, and product quality) of the dynamic system, the number of lost orders due to the inability of the agent to quickly accommodate changes and to attract new orders, etc. The agent behaves in a manner as to increase the fitness of the system that the agent belongs to locally, regionally and globally. To the extent that fitness criteria are shared, the aggregate group of agents tends to act in a collaborative fashion.

Each agent maintains this metric for its "local" fitness, as well as the "regional fitness" of the agent along with its immediate suppliers and customer. The following formulas which we use to calculate the fitness levels consider "0" to be optimal fitness. The *gap* measurement is mentioned above, and described in section 2-2.2. The *Δproduction* is simply the change in production from the previous cycle to the current cycle.

$$localFitness = |gap| + |\Delta production|$$

$$regionalFitness = \left(\sum_{i=1}^{r} localFitness(i) \right) / r$$

r = "number of <u>connected</u> customers and suppliers + 1"

The fitness function plays an important part in the evolving internal model. An agent can change its internal model based on its fitness levels, and the current state of the network. Agents monitor their fitness criteria with the use of fitness and gap functions to trigger the bounded control adjustment.

2-1.3 Internal Model

The internal model describes the reactive and proactive logic that the agent follows to determine its behavior. This model acts as the mental model or schema that the agent uses at any given moment to interpret its behavior and its external environments, and to generate the control inputs to the dynamic systems and the interactions with other agents. Hence, the internal model consists of the intra-agent control strategy in the controller and the performance assessment method. A part or the entirety of the internal model can be shared among a group of agents (e.g., shared norms, values, beliefs, and assumptions that make up the agent's internal model) or may be highly individualistic. The internal model is subject to evolution through learning and adaptation.

At any given time, the internal model of the agent can be built up by selecting any combination of building blocks associated with that agent. Decisions can then be made based on negative (achieving the goal set by the focal organization) and positive feedback (deviation from the goal set by the focal organization by having the focal organization accept this new goal). After making decisions, an agent can observe the direct effect of the decision on its fitness levels, and make adjustments accordingly.

The agents in our simulation select rules from the building blocks based on the homogeneity of the network and the level of information sharing between agents. These concepts are described in detail in section 2-2.2.

2-1.4 Building Blocks

The building blocks are sets of control and adaptation rules, algorithms and strategies from which the agent can choose to compose its internal model at any time. These may include control-theoretic algorithms, optimization algorithms, genetic algorithms, heuristic rules, and performance assessment methods. For example, agents are more cooperative under centralized control (top-down command, deterministic planning and re-planning in response to real time events, negative feedback control, proactive logic, the focal organization selects suppliers at all tiers, sets the unit prices and production levels throughout the supply network, service level, type of contract, and resource commitment of each agent). Agents are less cooperative under distributed control (bottom-up synchronization, positive feedback control, distributed decision making, reactive and adaptive logic, offer and counter-offer, accept or reject, all decisions involve only two parties: the upstream agent and the downstream agent). Levels of cooperation apply to issues of price, service level, resource commitment, and the selection and management of suppliers.

The building blocks included in our simulation include control strategies for determining production levels based on current level of homogeneity and information sharing within the network. The control strategies (described below) refer to network types and information sharing levels, which are described in section 2-2.2.

NETWORK TYPE	INFORMATION SHARING LEVEL	IF CONDITION:	SET PRODUCTION TO:
homogeneous	None	|gap| > 15 for 3 cycles	production – avgGap
	Regional	|customerGap| > 15 for 3 cycles	production - avgCustomerGap
	Global	|endGap| > 15 for 3 cycles	production – avgEndGap
semi-hetero & heterogeneous X={1..30} Y={1..6}	None	|gap| > X for Y cycles	production – avgGap
	Regional	|customerGap| > X for Y cycles	production - avgCustomerGap
	Global	|endGap| > X for Y cycles	production – avgEndGap

Table 2-1 Control strategies to set production level

2-1.5 Dynamic System

The dynamic system consists of mainly processes representing orders for finished goods, and resources taking processes and producing finished goods. Many firms satisfy orders by taking finished goods from the inventory, and then filling up the inventory through production using resources. Hence, material flows in the dynamic system may well be represented through the inventory level rather than the process flow of orders on resources. Our model of a supply network enterprises simplifies the representation of process flows on resources by applying the PUSH/PULL method of inventory control where orders are filled using inventory, and inventory is then filled using production. The inability to fill an order due to lack of sufficient inventory is reflected in the agent's fitness function.

2-1.6 Connectivity

The connectivity of the agent with other agents may manifest through material flows, information flows (e.g., the sharing of inventory information), and knowledge flows (e.g., the sharing of the internal model and/or fitness function). Different types and densities of connectivity determine different degrees of inter-agent control. For example, when there exists only material flows between a focal organization and one of its suppliers, the degree of inter-agent control is low. When there also exists information flows and even knowledge flows, the degree of inter-agent control between the focal organization and the supplier is high.

Agents in our simulation experiments are connected to one customer agent, and multiple supplier agents in a tree-like fashion. We present the details of this structure in section 2-2.1.

2-2 Network Design

In our experiments, we investigate different types of networks. Each of these networks is based on the same physical network structure. The variations introduced in this section represent variations on the state of the supply network as opposed to the physical positioning of the agents within the network.

2-2.1 Physical Structure

We choose a real supply network enterprise to determine the structure of our simulation model. This is a fixed network structure using a supply network with twenty-nine agents. The physical structure of the network maintains a hierarchy specified by the focal organization, followed by successive tiers of suppliers. During simulation, demand functions (described in section 2-2.2) determine the initial customer demand to place on the focal organization. This demand then propagates through the network in such a way that the demand placed on a supplier agent is equal to the production level of that agent's customer. This demand/production relationship is formed iteratively from the focal organization down to the farthest supplier tier.

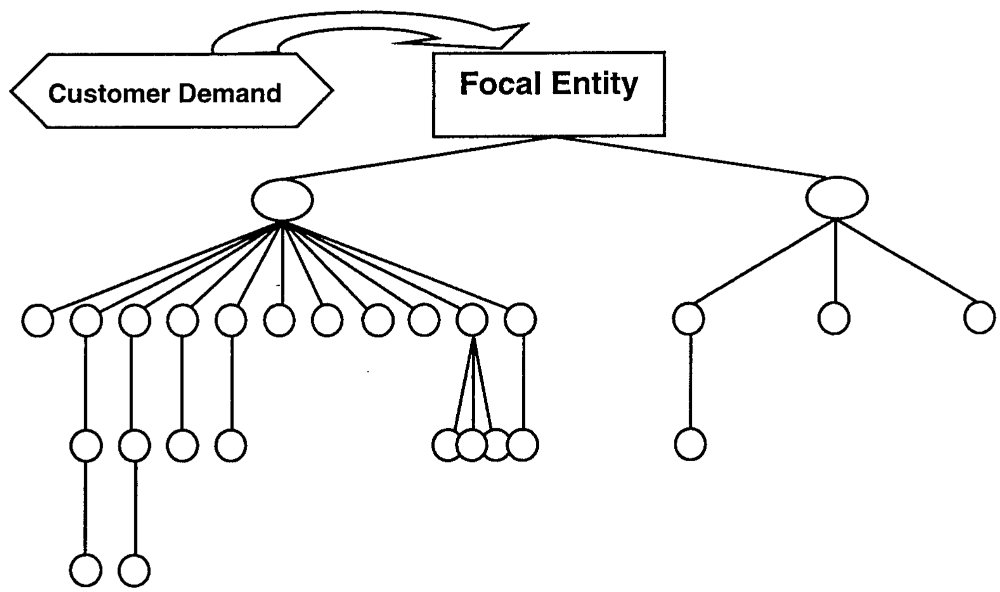

Figure 2-2 The structure of the supply chain enterprise used in the experiments.

The state of the network at any given time is defined by the type of the network (described in section 2-2.2) in conjunction with the network's fitness levels (described in section 2-2.3).

2-2.2 Network Types

At any given time, the state of the supply network enterprise can be given by its type and fitness levels (described in the next section). In this section, we define the type of a network.

There are four factors used to describe the type of a network. These factors are outlined in the following table.

FACTOR	POSSIBLE VALUES
Homogeneity	[homogeneous I semi-hetero I heterogeneous]
Level of Autonomy	[low I high]
Level of Information Sharing	[none I regional I global]
Market Condition	[stable I increasing I decreasing I volatile I seasonal]

Table 2-2 Factors that describe network types.

The homogeneity variable determines the extent to which agents within the system act in a similar fashion. In a homogeneous network, every agent has identical building blocks that it uses to build up its internal model, and each agent has the same initial production level. At the semi-hetero level, agents still have the same initial production level, but now have varying building blocks. The building blocks we use dictate when an agent will trigger a change in its dynamic system, and how much of a change will occur. These building blocks are described in Table 2-1. In the homogeneous system, the agents all modify their production values under the `same circumstances (if gap>15 for 5 cycles). In a semi-hetero network, the values 15 and 5 are replaced by randomly generated values in the ranges {1..30} and {1..6} respectively. The pre-determined randomly generated values used in our experiments are shown in the following table.

AGENT	X	Y	AGENT	X	Y	AGENT	X	Y
0F	6	3	2H	5	3	3D	29	2
1A	15	4	2I	13	3	3E	1	2
1B	6	5	2J	2	3	3F	4	5
2A	10	1	2K	13	3	3G	26	3
2B	8	1	2L	22	4	3H	28	5
2C	28	3	2M	16	4	3I	23	5
2D	10	2	2N	15	1	4A	2	3
2E	30	4	3A	1	6	4B	8	3
2F	3	3	3B	10	2			
2G	25	6	3C	12	6			

Table 2-3 Randomly generated values for heterogeneous network functions.

In this and the following table, the name given to an agent includes its level in the network structure (number), and the order in which it appears in that level (letter). The X and Y columns correspond to the X and Y values of the production functions shown in section 2-1.4. In a heterogeneous network, the building blocks remain varied, and each agent now has a unique, randomly generated, initial production level in the range {500-1,500} as shown in the table below.

AGENT	PRODUCTION	AGENT	PRODUCTION	AGENT	PRODUCTION
0F	1262	2H	743	3D	1016
1A	998	2I	1350	3E	718
1B	1377	2J	1230	3F	937
2A	745	2K	792	3G	1311
2B	1306	2L	1336	3H	841
2C	558	2M	1462	3I	1063
2D	703	2N	905	4A	886
2E	562	3A	1407	4B	503
2F	1428	3B	540		
2G	960	3C	1039		

Table 2-4 Randomly generated initial production values

The level of autonomy determines how much freedom is given to individual agents within the system to make their own decisions. Under high autonomy, the focal organization sets all the production values throughout the network using a production function based on the type of network. In low autonomy, each agent uses the production functions to determine its own production level, which is also based on the type of network. Therefore, the level of autonomy simply determines whether the network is operating under centralized or distributed control.

The level of information sharing within the network dictates how much a supplier knows about its customer. In our simulation experiments, we model the information sharing by giving the agents varying levels of access to upstream agents' demands. The three levels of information sharing are reflected in the three gap functions mentioned at the beginning of this chapter and restated here for convenience:

INFORMATION SHARING LEVEL	IMPLICATIONS	RESULTING GAP FUNCTION
None	Agent is only aware of its own demand	*production(i) - demand(i)*
Regional	Agent is aware of its demand and its immediate customers demand	*production(i) - demand(customer)*
Global	Agent is aware of its demand, its immediate customers demand, and the end customer demand imposed on focal organization.	*production(i) - demand(endCustomer)*

Table 2-5 Result of information sharing on agent knowledge (the gap functions).

These gap functions are a reflection of the knowledge an agent possesses of its environment. At each level, the agent has access to the information available at the level it is currently in, and any levels above the current level on the table. However, the agent cannot access information on a level in the table that is lower than the level of the network in which the agent belongs.

The final factor that determines the type of network is the market condition. The market conditions determine the value of the end customer demand, which is imposed on the focal organization. There are five functions to describe the five different market conditions. The end customer demand is based on a total possible range of {0..2,000} to standardize the experiments. The demand functions are listed in the following table.

MARKET CONDITION	DEMAND FUNCTION	DEMAND RANGE
Stable	$a+e$	900..1,100
Increasing	$a[1+(t/\#Cycles)]+e$	900..2,100
Decreasing	$a[1-(t/\#Cycles)]+e$	1,100..0
Volatile	$a+E$	0..2,000
Seasonal	$if\ \frac{1}{3}\#Cycles < t < \frac{2}{3}\#Cycles, then = 2a$ *else = a*	1,000 or 2,000

Table 2-6 End customer demand by market.

In the table above:

- *'a'* is the initial production for each agent.
- *'t'* is the current cycle [1 .. 100,000].
- *'e'* is a randomly generated number [-100 .. 100]
- *'E'* is a randomly generated number [-1,000 .. 1,000]

The output of these demand functions is shown in the following charts.

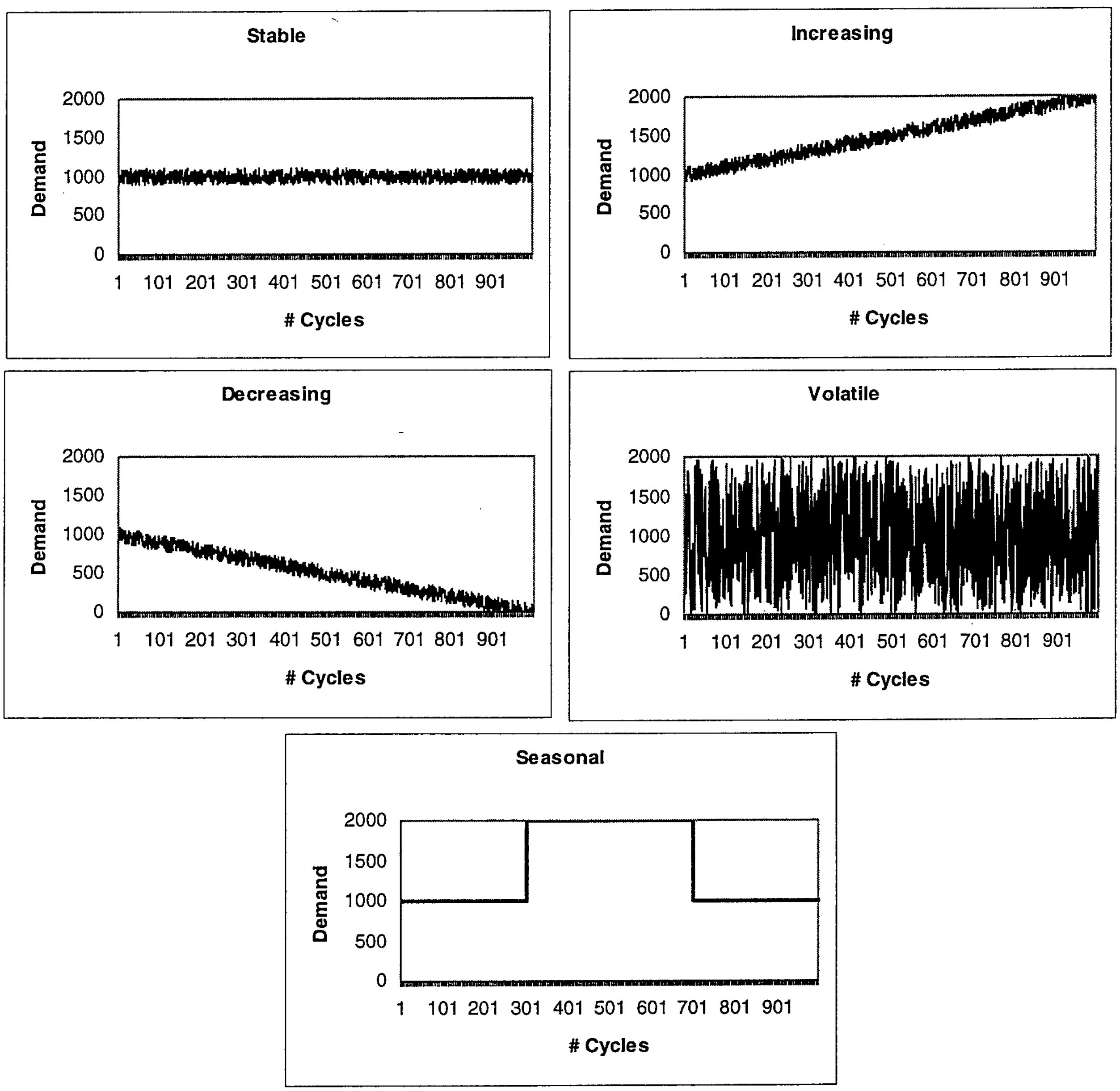

Figure 2-3 Output of customer demand functions.

2-2.3 Fitness Metric

The fitness metrics used for the network have constructs similar to those used for the individual agents, described in section 2-1.2. Similar to the individual agent's fitness functions, the following fitness functions consider zero optimal.

$$globalFitness = \left(\sum_{i=1}^{n} \left| endGap(i) \right| \right) / n$$

n = *"number of agents in network"*

$$networkFitness = \left(\sum_{i=1}^{n} localFitness(i) \right) / n$$

Here, the global fitness is a reflection of how well the agents in the network collectively meet the needs of the end customer. The network fitness is simply an average of the local fitness's of all agents in the network.

These two fitness metrics are the key to understanding the dynamics of the system as a whole. Although it would be beneficial to analyze each agent's local and regional fitness on an individual basis, in this project, we limit our analysis to the global and network fitness levels of the entire network and leave more in depth analysis to further study.

Chapter 3: Design of Simulation Experiments

We will run a series of simulation experiments that investigate the emergent behavior of a supply network enterprise under various conditions of three factors: inter-agent control method, supply network structure, change in internal dynamics and external environment. We will examine both the performance aspect and the structural aspect of the emergent behavior of the supply network enterprises. The inter-agent control methods will vary from only material flows, to material flows and information flows, and finally to material flows, information flows, and knowledge flows. For this one-year project effort, our network has a static physical structure. In future experiments, the supply network structures will differ in the number of tiers of suppliers for a focal organization, number of suppliers in each tier, the involvement (full or partial) of resources from a supplier in the supply network enterprise, and so on. Changes in internal dynamics and external environment will be set through the rate and type of changes within individual agents and from the external market environment.

In summary, in simulation experiments, we will investigate the following factors:

1) Inter-agent control methods for different degrees of agent autonomy such as sharing of state information among agents, conformance to a common fitness function by agents, and enforcement of the same internal model among agents; and
2) System structures determined by types and density of inter-connections among agents through material and information flows.

The design of these simulation experiments is based on a combination of current design techniques in supply networks, complex adaptive systems, multi-agent systems, production-distribution models, dynamic networks, self-organizing structures, networks exhibiting emergent behavior, distributed control systems, and discrete event systems. [1-10]

3-1 Initial Setup

In chapter 2, we defined state variables to describe the simulation system. These variables are summarized in the following table.

VARIABLE TYPE	VARIABLE	POSSIBLE VALUES / IMPLICATIONS
Independent (Network Level)	homogeneity	[homogeneous I semi-hetero I heterogeneous]
	level of autonomy	[low I high]
	information sharing	[none I regional I global]
	market condition	[stable I increasing I decreasing I volatile I seasonal]
Independent (Agent Level)	Role	[focal I 1Tier I 2Tier I ... I nTier]
	Internal Model	Defined by current set of building blocks.
	Building Blocks	Set of functions based on type of network.
	Connectivity	Number of connected suppliers.
Dependent (Network Level)	customer demand	Based on market conditions.
	global fitness	Average endGap of all agents in network.
	network fitness	Average local fitness of all agents in network.
Dependent (Agent Level)	production level	Based on homogeneity, autonomy, & info sharing.
	local fitness	Based on gap.
	regional fitness	Average local fitness of all agents in region.

Table 2-7 Summary of simulation state variables

Our simulation contains four independent variables. The number of possible values for these variables are: three, two, three, and five respectively. This gives us a total of 90 simulation states. Throughout the study, we narrow our focus to those states of greatest interest.

In those experiments where the agents' initial production value is the same (homogeneous and semi-hetero networks), we set the initial production to 1,000. Experiments are originally run for 1,000 cycles. We do some minor checking of differing cycle times as shown in the results in chapter 5.

3-2 Running the Experiments

Each of the simulation sessions processes a set of experiments based on the types of networks under investigation. Each experiment in the set is run for 1,000 cycles. Some experiments are run for 2,000 and 10,000 cycles to further investigate effects of simulation length. Each cycle of a simulation experiment consists of the following five (six) steps:

1. Calculate end customer demand
2. Process order through network
3. Record 3 dependent variable values per agent
4. Record global fitness and network fitness
5. Calculate next cycle's production level by agent
    a. If autonomy is low – focal sets production level across network
    b. If autonomy is high – each agent determines own production level
6. Calculate next cycle's control strategies using adaptive logic for each agent.

    This step is only used in Class 3 simulations (see below).

Following the completion of an experiment, the network and its agents are reset to their respective initial states before the next experiment in the set (or set of experiments) is run.

To single out specific research aims, we break the testing down into three experiment classes. For each class, changes are made to the experiment set up based on observed simulation results. Each class removes uninteresting aspects from the previous class, changes some aspects to refine results, and adds new aspects to further investigate interesting phenomenon. The classes are divided as follows:

Class 1 – Investigates the effects of market conditions, information sharing, autonomy, homogeneity (limited to the homogeneous and semi-hetero levels), and the costs associated with changing production levels.

Class 2 – Investigates the effects of noise present in customer demand functions, length of simulation (# of cycles), homogeneity (limited to the semi-hetero and hetero levels), and standardizing the change in production costs.

Class 3 – Investigates the effects of adding adaptive logic to allow agents the ability to adjust their control strategies.

These experiment classes are further defined in the following three sections.

3-2.1 Class 1 Experiments

In this class of experiments, all settings are as described previously. This class is used as a basis to determine which factors contribute greatest to the overall fitness of the system. During this class of experiments, we limit investigation into the homogeneity of the system to purely homogeneous and semi-hetero network types. We also concentrate our efforts on determining the magnitude of the market effect on the network performance. Furthermore, we investigate the effects of change in production costs factored into the fitness functions.

3-2.2 Class 2 Experiments

Because of class 1 analysis, further described in section 5-1, we make the following changes in class 2:

- Remove homogeneous network type
- Remove volatile market condition
- Remove change in production cost from local fitness function

We enhance the investigation of the homogeneity of the network by adding a heterogeneous network type to compare with the semi-heterogeneous network. We further investigate the effects of market conditions by removing the 'noise' from the demand by market formulas. This is accomplished by simply removing the random *e* and *E* from the first four market conditions. To balance the effect of removing costs of change in production levels, we alter the production functions to increase or decrease the production by a factor of ten each time the production requires a change.

Semi-hetero and heterogeneous networks: (X&Y are random) X={1..30} Y={1..6}

if |workingGap| > X for Y cycles

newProduction = production +/- 10This modification increases the number of cycles needed to catch up production, which "magnifies" the inner workings of the network. These new formulas make better sense for a network with no production costs since now production can only change by a pre-selected amount.

In addition to the above changes, we increase the length of a simulation by running the experiments for one, two, and ten thousand cycles. These results are described in section 5-2.

3-2.3 Class 3 Experiments

Because of class 2 analyses, further described in section 5-2, we keep all changes initiated in class 2. We also eliminate the experimental cycle length of 1,000 cycles for reasons given in section 5-3. Further changes in this class investigate the effect of adding adaptive control logic within individual agents. This control logic allows an agent to change its internal model by altering the control strategies within its current set of building blocks following each cycle depending on the following logic:

If newGap >= oldGap for 5 cycles; decrement X & Y

If newGap < oldGap for 5 cycles; increment X & Y

In these formulas, X and Y are from the production logic formulas given in table 2-1 for semi-hetero and heterogeneous network types. The gap is determined by the level of information sharing as described in table 2-5. This logic tells each agent when to trigger a control change. Each time an agent triggers a control change, the X and Y values are either incremented or decremented by one accordingly. Further investigation into heterogeneous complex adaptive networks would involve allowing varying adaptive logic across the network. This one-year project focuses on the effects of all agents sharing the same adaptive logic.

3-3 Collecting Data

Following each cycle, the dependant variables associated with the network and the individual agents that are updated include all fitness levels, agents' production levels, and agent control strategies when triggered. The latter only applies to class 3 experiments. The updating occurs in a sequential fashion such that any variables, which rely on other variables, are updated following the variables that they rely on as follows:

- As order is processed through network, for each agent:

- ✓ Set gap levels for all three gap functions
- ✓ Calculate local fitness for cycle

- Following order processing:
  - ✓ For all agents – set regional fitness
  - ✓ Set global fitness
  - ✓ Set network fitness
  - ✓ Set production level for all agents based on level of autonomy
  - ✓ If class three experiments, set control strategies if logic triggers a change for this cycle.

All data is collected in two formats for use in the two different types of data analysis described in chapter 4.

Chapter 4: Statistical Data Analysis

We investigate the non-linear time-series analysis techniques from chaos and dynamical systems theories and multivariate statistical analysis techniques, and transform them into the scalable, applicable techniques for analyzing the emergent behavior and temporal patterns of the supply network system using the data obtained from the simulation experiments. [7-14] In section 3-3, we discussed the data collection process for each simulation experiment. This collected data is represented in multiple formats that correspond to the desired method of analysis. The two main analysis goals for this project include the performance of the system, and detecting emergent behavior in the system.

We investigate how to represent the simulation data from the model of a supply network enterprise in a form that is acceptable to these analytical techniques, and how to reform and advance these analytical techniques so that they are scalable for real-time data analysis. Throughout the analysis process, techniques for analyzing data, as well as techniques for appropriately collecting data, are refined to enhance the validity and comprehension of each metric.

4-1 Measuring Performance

Performance is measured through the analysis of inherent patterns within the collected simulation data. To analyze the performance of the entire supply network enterprise, we focus our attention on the dependent variables: production, global fitness and network fitness. The data output from the simulation process is formatted in two ways to enable the import of data into graphing software to perform visual observations as well as the ability to import the collected data into statistic software for further analysis, including ANOVA analysis and Tukey HSD tests. The measure of performance allows the determination of control techniques and network types that enhance performance versus those that worsen it. Accurate analysis of the performance of the supply network enterprise serves to direct research into the appropriate areas of communication and control necessary to improve the performance of the system.

4-1.1 Visual Observations

For visual observations, the data is imported to a graphing software package, where it is plotted for analysis. For each experiment, we create production and fitness plots. The production charts include the production level of each agent in the network over the cycles in the simulation experiment. With these plots, we can visualize the effect of information sharing, homogeneity, autonomy, and control functions on the production of the agent system. Global fitness and network fitness plotted together for each experiment allow observation of the comparison and interaction of these two metrics. Over a set of experiments, we create separate plots for global and network fitness. Each chart shows how the respective fitness level varies under differing network types and simulation classes.

Visual observations provide a dual purpose. They allow the confirmation of known or suspected system behavior, and therefore give assurance that the simulation software is behaving as expected. Furthermore, visual observations are an easy way to detect areas of study that are deserving of further attention, as well as those that do not appear to contain any further enlightening discoveries. Visual observation is used as a first step in our data analysis for the benefit of both of these principles.

4-1.2 ANOVA Analysis

An analysis of variance (ANOVA) is used to test for statistically significant differences in means. The testing is based on the partitioning of the variances. [10] The results of analysis show us the magnitude (F) of the relationships of variables, or differences of means, and the statistical significance (p-value) of each result. The p-value of a result is a measure of how significant the observed relationship or difference really is. This value is the probability that the observation occurred by pure chance. In our study, we only consider observed results significant if the corresponding p-value is $< .05$. ANOVA allows us to determine which of the four factors given in table 2-2 contribute greatest to the observed effect on the different fitness levels.

Because the supply network enterprise is multivariate, we use multi-factor ANOVA to produce tabular and graphical comparisons of the effects of the multiple factors on a single dependent variable. We use graphs produced from ANOVA results to show the 4-way interaction between factors for both the global and network fitness metrics in chapter 5. Other results show us the effects of 2-way and 3-way interactions for more detailed observations of the relationships between factors.

For each of our experiments, the results include the level of each fitness metric for every cycle in the experiment. For example, after 1,000 cycles, we have 1,000 local fitness points for each agent in the network, 1,000 global fitness points, and 1,000 network fitness points. We process the ANOVA analysis for the local fitness of every agent, and the global and network fitness metrics. For this phase of the project, we focus on the global and network fitness levels. For each of these fitness metrics, we produce a summary of all effects that shows 1-way, 2-way, 3-way, and 4-way interactions between the four factors for that fitness metric. This summary includes the F and p-levels as described above. Next, we create a graph of the 4-way interaction to visualize the interaction between factors. Finally, we process Tukey HSD tests (described below) for all statistically significant results in the summary table to further analyze the effects of the factors and their interaction with each other.

Results for each experiment are compared with other experiment results within the same class to identify important factors and interesting behavior. The results of a class of experiments are then compared to another class for identifying behavioral changes that occurred by instituting changes to the simulation. This process allows us to refine our simulation and analysis techniques.

4-1.3 Tukey HSD Testing

A post-hoc comparison of means is given by the Tukey honest significant difference (HSD) test. This test allows the grouping of means to see which groups are particularly different from each other. The use of post-hoc comparison techniques is preferred over other techniques, such as the t-test for independent samples, because they take into account that more than two samples were taken. This prevents the possibility of reported probability levels overestimating the statistical significance of the mean differences. With the Tukey tests, we can see how the factors work together in producing the statistically significant effects observed in the ANOVA testing process. Many of the observations and discussions given in chapter 5 result from analysis of the Tukey HSD tests results for each

The format of the Tukey HSD test allows processing tests with 1-way, 2-way, 3-way, and 4-way interactions. This way, we can run a Tukey test on any of the observed statistically significant interactions found in the ANOVA summary described above. The output is a table with rows indicating all possible combinations of factors. These rows are ordered such that the combination resulting in the smallest mean is first, and the combination resulting in the largest mean is last. Through this format, we can look at a Tukey result and see immediately which combination of factors gives the best and worst performance values for the network. Furthermore, the Tukey HSD results break up the rows into groups, which have means that are statistically similar when compared to the other combinations of factors. With this, it is easy to identify which factors contribute the most to a major change in performance level, and which factors only contribute to minor performance changes.

4-2 Detecting Emergent Behavior

We consider a number of non-linear time-series analysis techniques such as the delay-coordinate embedding technique to reconstruct the phase space from time-series data, estimate the dimensionality of the system state via the correlations dimension, and thus detect emergent states of the system, as well as multivariate statistical analysis techniques such as clustering and classification techniques. [9-14] The importance of detecting emergent behavior is abundant. In one case, if the emergent behavior of the network is producing damaging effects to the overall performance of the system, control techniques can be applied to drive the network out of the emergent state. In another case, if a network is exhibiting emergent behavior that is proving beneficial to system performance, alternate control strategies (or perhaps the absence of control), can be applied to keep the system in the emergent state. Furthermore, in a proactive way, if we can find the combinations of factors that drive a network into an emergent state, we can either prevent or encourage this combination of factors depending on whether the emergent state is considered good or bad for the network.

4-2.1 Visual Observations

As with the performance measurement techniques, we begin by making visual observations in an attempt to detect emergent behavior in the system. These observations assure us that our simulation is behaving as expected and that the data collected from the system is adequate to subject to more complex emergent behavior analysis techniques. To detect the presence of emergent behavior, we plot the production levels of each agent over the course of a simulation and compare the series. If the agents' production levels appear sporadic and undeterminable, we predict that the simulation will not exhibit emergent behavior. However, if the agents' production levels seem to converge at any point, we guess that the simulation will not exhibit emergent behavior. With the knowledge gained from these visual observations, we can continue our study into the emergent behavior of the network through more sophisticated techniques.

We observe the output of each experiment with respect to global and network fitness. In the cases where the output appears chaotic, we determine that chaotic time series analysis is appropriate.

4-2.2 Chaotic Time Series Analysis

Chaotic time series analysis is used to detect emergent behavior in a system by identifying the deterministic origin of a time series with chaotic underlying dynamics. By estimating the dimensionality of the stochastic process, it is possible to detect a chaotic time series in the output of the process. In chaotic time series analysis, delay coordinates are commonly used to reconstruct an image of a dynamic system. [11] The correlation dimension (D_2) of the image of an attractor in the original dynamic system can be estimated by a technique presented in [11-14] which uses the embedding dimension and delay time coordinates to estimate D_2. This technique applies the selection of appropriate values for the embedding dimension and delay time to the Grassberger-Procaccia algorithm.

This algorithm evaluates D_2 using the probability that a randomly chosen pair of points will be separated by a distance less than ε on the attractor [11-14]

$$C_N(\varepsilon) = \frac{2}{N(N-1)} \sum_{j=1}^{N} \sum_{i=j+1}^{N} \Theta(\varepsilon - \|x_i - x_j\|)$$

where Θ is the Heaviside function [$\Theta(x)=1$ if $x \geq 0$, and $\Theta(x)=0$ otherwise]

and $\|x\| = max\{\|x_i\| : 1 \leq i \leq m\}$, *where m is the embedding dimension*

Then the correlation dimension is given by:

$$D_2 = \lim_{\varepsilon \to 0} \lim_{N \to \infty} \frac{\log C_N(\varepsilon)}{\log(\varepsilon)}$$

For our application, we designate the correlation sum as $C_N(\varepsilon,\tau,m)$. [11] Where τ is a constant time-delay selected by observing a graph of the stochastic process output, and m is a variable representing the embedding dimension. We plot the slope of the linear portion of D_2 over increasing values of m to detect emergent behavior within the supply network enterprise. If emergent behavior exists in the network, then for some dimension m, the slope of D_2 is the same for all $x \geq m$. Typically, the slope of D_2 increases with m until this plateau is reached. [11]

In detecting emergent behavior, we run this algorithm on the global fitness and network fitness outputs of each simulation experiment that exhibits chaotic time series output as observed in the visual observations of 4-2.1.

Chapter 5: Results and Discussion

The results presented here are separated by experiment class. The classes of experiments are described in section 3-2. These classes allow us to single out specific areas of interest for differing research aims. Within each sub-section of this chapter, we describe results relating to key areas of investigation for that class, changes made to the simulation model from previous classes, and results that may warrant changes in future classes.

5-1 Class 1 Experiments

Our initial set of experiments show that the effect of a volatile market is highly significant. The volatile market shows significantly worse performance levels than the other four markets (stable, increasing, decreasing, and seasonal) for both global and network fitness. For the global fitness, market level is the only significant factor. We determine that fitness is driven by the market due to a cancellation effect (when one tier has poor fitness, another has good fitness).

For network fitness, all four factors are significant. The significant factors, when analyzed unaccompanied (1-way interaction), which result in the best performance are homogeneous; low autonomy; and global information sharing. We observe that a homogeneous, high autonomy network with global information sharing can achieve the same effects as a homogeneous, low autonomy network. Furthermore, a semi-hetero, high autonomy network achieves best fitness when each agent reacts only to its immediate customer (no information sharing). Another significant observation is that increasing the level of information sharing among autonomous agents improves network fitness in homogeneous networks, and worsens it in semi-hetero networks.

In the graphs that follow, the first four pairs of graphs represent global fitness, and the next four pairs represent network fitness. The pairs represent autonomy (low and high), and there is one pair for each of four market conditions (stable, increasing, decreasing, and volatile). The y-axis of each graph is fitness level, and the x-axis is the information sharing level (none, regional, global). The lines represent the homogeneity of the network (homogeneous and semi-hetero).

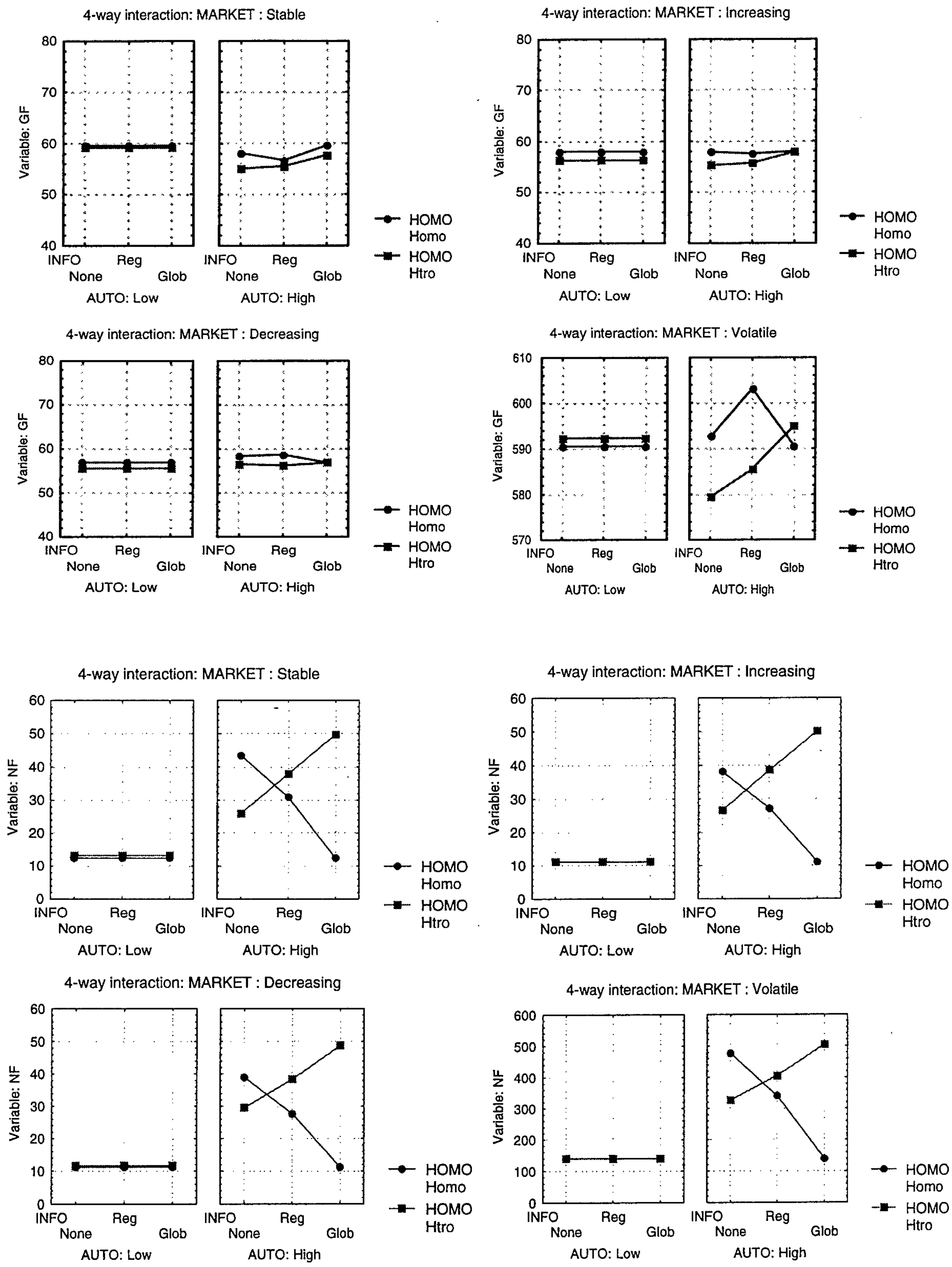

Figure 2-4 Global and network fitness in class 1 experiments.

Observe that for each market condition, the underlying patterns of variable interaction are similar, however, the values of the fitness levels are significantly higher (worse) in the volatile market. The graphs also clearly identify the cross-relationship between homogeneity and information sharing. Chaotic time-series analysis shows that this network is a purely stochastic system exhibiting no emergent behavior.

In the seasonal market, for global fitness all factors except homogeneity are significant. The best significant 1-way factors are seasonal and low autonomy. In the network fitness, all factors are significant. The best significant 1-way factors are seasonal, homogeneous and low autonomy. Information sharing improves both fitness measures in the seasonal market for homogeneous and semi-hetero autonomous networks.

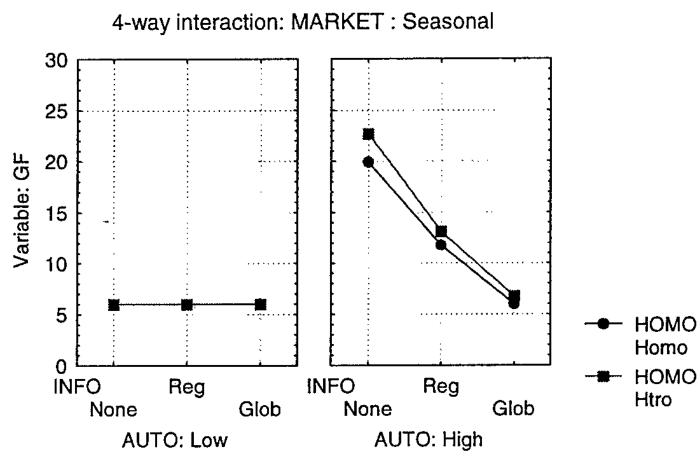

Figure 2-5 Global fitness is a seasonal market.

For all simulations, low autonomy provides better performance overall. The seasonal market with its absence of noise provides the best fitness levels. Observe also that the cross-interaction between homogeneity and information sharing does not exist without the noise present in other market conditions.

Recalling that our fitness levels take into account the agent's ability to meet customer demands, as well as the cost of changing fitness levels, we investigate the effect these costs have on the overall fitness.

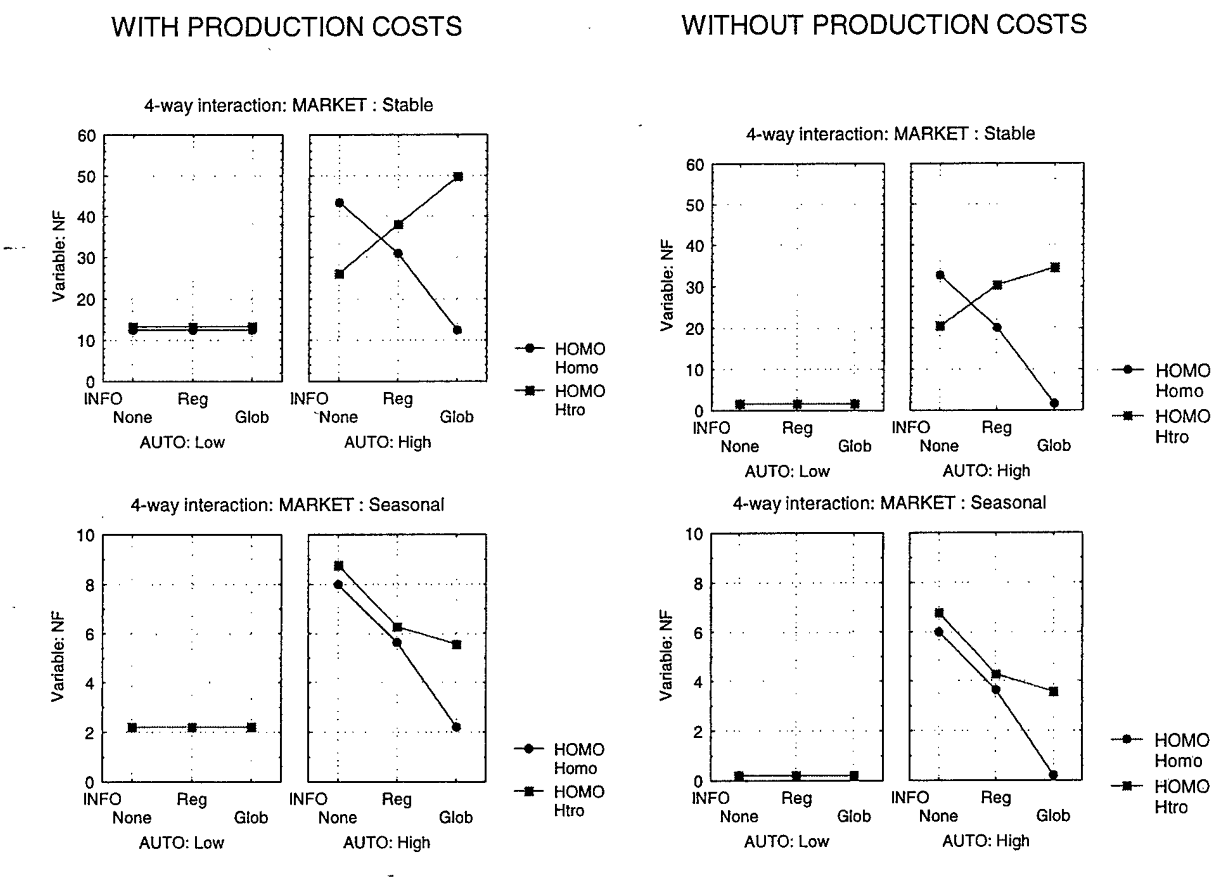

Figure 2-6 Effects of production costs on the fitness levels.

As shown in this chart, the effect of calculating the change in production costs into the fitness metric does not have a significant effect on the overall performance of the network.

5-2 Class 2 Experiments

The experiments in this class follow the changes to the model outlined in 3-2.2. The results of these modifications are as follows:

- Global Fitness
    - Increased information sharing improves fitness in an autonomous network.
    - Best significant 1-way: low autonomy; global info sharing
- Network Fitness

- Fitness worsens or stays the same with increased information sharing in an autonomous network.
- Best significant 1-way: low autonomy; no info sharing

• Both fitness measures

- All factors are significant.
- Seasonal market shows the worst performance levels.

In the graphs that follow, the first set of four graphs is global fitness and the second set is network fitness. In this example, we only look at the purely heterogeneous network to compare with semi-hetero graphs from the previous class of experiments. Therefore, the lines on the graphs now represent low and high autonomy so that we can measure the significance of this factor. The remaining parts of the graphs remain the same as the last set.

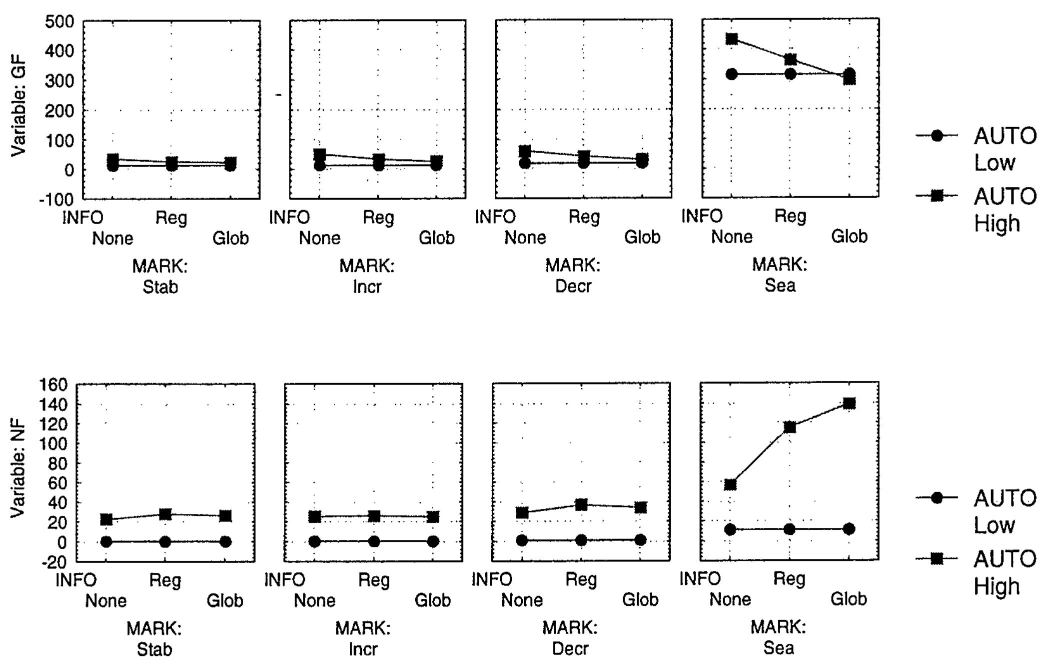

Figure 2-7 Global and network fitness in class 2 experiments.

From these graphs, we can see the significant effect of the changes outlined in section 3-2.2. In further experiments, we ran the simulation for 2,000 and 10,000 cycles. As the number of cycles increased, we find that the overall fitness levels improve (with very small changes between 1,000 and 2,000 and larger changes between 2,000 and 10,000). However, the overall Interaction between the factors remains the same. Additionally, the negative effect of information sharing on network fitness is decreased (positive in some markets) at 10,000 cycles.

With the removal of noise from the demand functions, we find that the output of our simulation experiments no longer has an underlying stochastic structure, which nullifies the need for detecting emergent behavior. We continue our in depth study into the workings of the supply network enterprise to complete this one-year effort with the aim to refine our model before reverting back to a more complex customer demand market. We save the further research into the analysis and control of emergent behavior to future study, and continue investigating our model.

5-3 Class 3 Experiments

In order to witness the effects of adding adaptive logic to the agents, we run the simulations for 2,000 cycles. As observed in class 2 experiments, this will not degrade our result comparisons with the other classes. The charts showing the global and network fitness of this class are set up in the same way as the class 2 experiment charts. As with the class 2 experiments, all factors are significant for both global and network fitness measures.

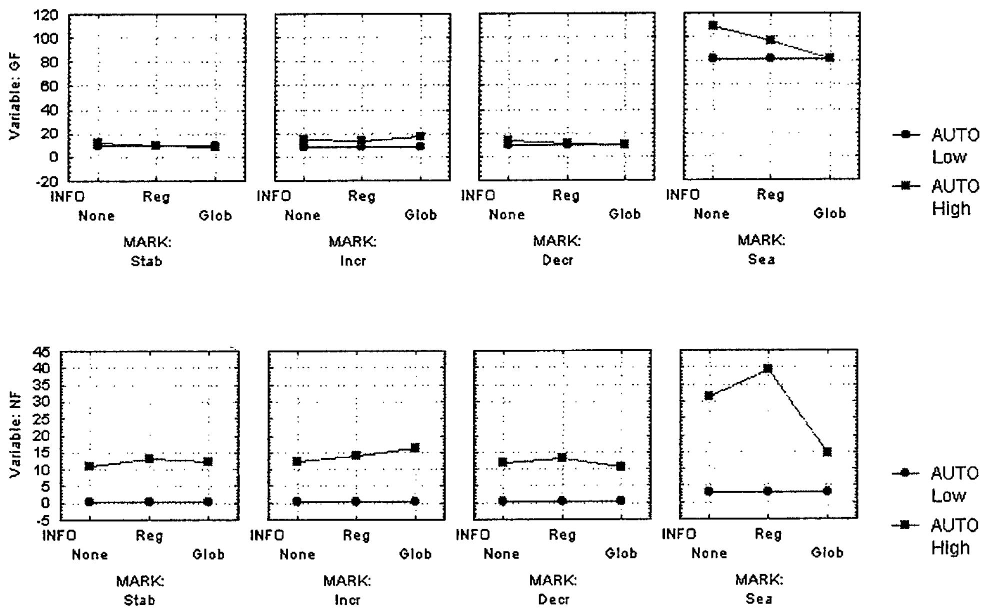

Figure 2-8 Global and network fitness in class 3 experiments.

We observe from these charts, that the overall pattern of variable interaction remains the same as in class 2 experiments, with the exception of the network fitness in a seasonal market. However, the level of fitness is significantly lower (better) in all cases. This follows from the scale along the y-axis. The interesting observation here is that the changes made in class three simulations have a significant positive impact on the network fitness level in a seasonal market with high autonomy, and global information sharing.

Chapter 6: Conclusions

In this project, we have shown a method for designing a model of a supply network enterprise. With this model, we use our simulation method to analyze the behavior of the supply network enterprise. Our analysis techniques prove creditable as shown in the simulation results of chapter 5. This one-year project has culminated in the design and implementation of a simulation model, the refinement of the model through simulation processing, the application of analysis techniques to understand the simulation results, and the resulting observations of a complex, dynamic supply network enterprise. To summarize our results, we discovered the following:

- ✓ Volatile markets have the most negative impact on the system performance.
- ✓ Low autonomy has the most positive impact overall on the system performance.
- ✓ A homogeneous, high autonomy network with global information sharing can achieve the same performance as a homogeneous, low autonomy network.
- ✓ Noise present in customer demand creates a cross-relationship between homogeneity and the level of information sharing.
- ✓ Performance does not always improve with increased information sharing in an autonomous network.
- ✓ With adaptive logic, agents can improve their fitness levels, thereby improving the performance of the entire system.

These observations provide a basis for observing the effects of our simulation model in chaotic environments.

References

1. Choi T, Dooley K, Rungtusanatham M. *Supply Networks and Complex Adaptive Systems: Control versus Emergence*. Journal of Operations Management 19: 351-366, Elsevier Science B.V., The Netherlands, 2001.
2. Swaminathan J, Smith S, Sadeh N. *Modeling Supply Chain Dynamics: A Multiagent Approach.* Decision Sciences, Volume 29 Number 3, Summer 1998.
3. Chang M, Harrigton, Jr. J. *Centralization vs. Decentralization in a Multi-Unit Organization: A Computational Model of a Retail Chain as a Multi-Agent Adaptive System,* Cleveland State University, September 1999.
4. Larsen E, Morecroft J, Thomsen. *Complex Behaviour in a production distribution model,* European Journal of Operational Research 119: 61-74, Elsevier Science B.V., The Netherlands, 1999.
5. Lin F, Huang S, Lin S. *The Effects of Information Sharing on Supply Chain Performance in Electronic Commerce,* National Sun Yat-sen University, Kaohsiung, Taiwan 804 R.O.C.
6. Periorellis P, Dobson J, *Modelling Dynamic Networks Advantages and Limitations of the O-O Approach,* Center for Software Reliability, University of Newcastle, U.K.
7. Banks J, Carson II, J. *Discrete-Event System Simulation,* Prentice-Hall, Inc., Englewood Cliffs, NJ, 1984.
8. Walker C, Dooley K. *The Stability of Self-Organized Rule-Following Work Teams,* Kluwer Academic Publishers, The Netherlands, 1999.

9. Kogut B. *The Network as Knowledge: Generative Rules and the Emergence of Structure,* Strategic Management Journal, 21:405-425, 2000.

10. Parunak H, VanderBok R. *Managing Emergent Behavior in Distributed Control Systems,* Industrial Technology Institute, Ann Arbor, MI.

11. Lai Y, Lerner D. *Effective scaling regime for computing the correlation dimension from chaotic time series,* Physica D 115: 1-18, Elsevier Science B.V., The Netherlands, 1998.

12. Lai Y, Osorio I, Harrison M, Frei M. *Correlation-dimension and autocorrelation fluctuations in seizure dynamics,* Arizona State University, Tempe, AZ, 2001.

13. Lai Y, Kostelich E. *Are spatiotemporal dynamics detectable from delay-coordinate embedding of scalar time series?* Arizona State University, Tempe, AZ, 2001.

14. Dhamala M, Lai Y, Kostelich E. *Analyses of transient chaotic time series,* Georgia Institute of Technology, Atlanta, GA, 2001.